

**Thuedi**

**Software Requirement Specification**

– Hanoi, January 2022 –

**Table of Contents**

[I. Overview](#_heading=h.gjdgxs) **4**

[1. Introduction](#_heading=h.30j0zll) 4

[2. System Functions](#_heading=h.ft3522ht4lrq) 5

[a. Screen Flow](#_heading=h.mad4zwods7bl) 5

[b. Screen Details](#_heading=h.fhviy9gte6gj) 5

[c. User Authorization](#_heading=h.pizf5rm9sh62) 6

[d. Non-Screen Functions](#_heading=h.3wxp05iku9v2) 7

[3. Common Requirements](#_heading=h.3znysh7) 7

[II. Functional Requirements](#_heading=h.2et92p0) **8**

[1. Authentication](#_heading=h.tyjcwt) 8

[User login](#_heading=h.3dy6vkm) 8

[User registration](#_heading=h.1t3h5sf) 8

[Reset password](#_heading=h.nou8p3a35z7s) 8

[2. Landing page](#_heading=h.4d34og8) 8

[Landing page](#_heading=h.jtb6d3k7j4ag) 8

[3. Management](#_heading=h.w71dp8fxsp4s) 8

[Admin page](#_heading=h.gih2nkxbjs2n) 8

[4. Post detail](#_heading=h.qjp7pfb19ic8) 8

[View Post](#_heading=h.e3fk5pid3nqm) 8

[View covid status near address](#_heading=h.qxnotec68xvm) 9

[Comment on post](#_heading=h.r9gal36qstkx) 9

[View services near address](#_heading=h.8fso6r3fzdvf) 9

[Report post](#_heading=h.qnrk0ysumd6t) 9

[5. Post management](#_heading=h.51gqx7f819h) 9

[Create post](#_heading=h.wz3cp7imjau2) 9

[Pay to create post](#_heading=h.vmwubw2h32oy) 9

[Update post](#_heading=h.x87c5o1wktao) 10

[Delete post](#_heading=h.hw14xdhu1gaw) 10

[6. Posts](#_heading=h.320qs3j0mc0q) 10

[Post list](#_heading=h.452f9qp4hfpa) 10

[Filter post](#_heading=h.lb6dlyuv6ez) 10

[Post list pagination](#_heading=h.z6popqbsegod) 10

[Post list breadcrumb](#_heading=h.38w6odns02n) 10

[7. Review area](#_heading=h.ntw8gjdrywd1) 10

[Show a review area](#_heading=h.hwdzpd1vvi0m) 10

[8. User profile](#_heading=h.mdyu6jownmu) 11

[User Profile](#_heading=h.fs2vb63msixw) 11

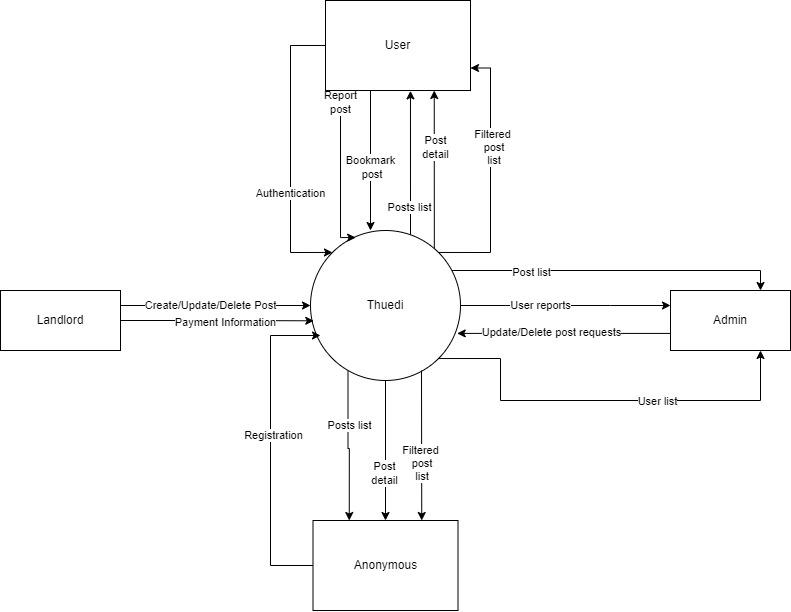
[Bookmark post](#_heading=h.1l1kvb5uluim) 11

[Bookmark management](#_heading=h.sutn4s7eyuo4) 11

# I. Overview

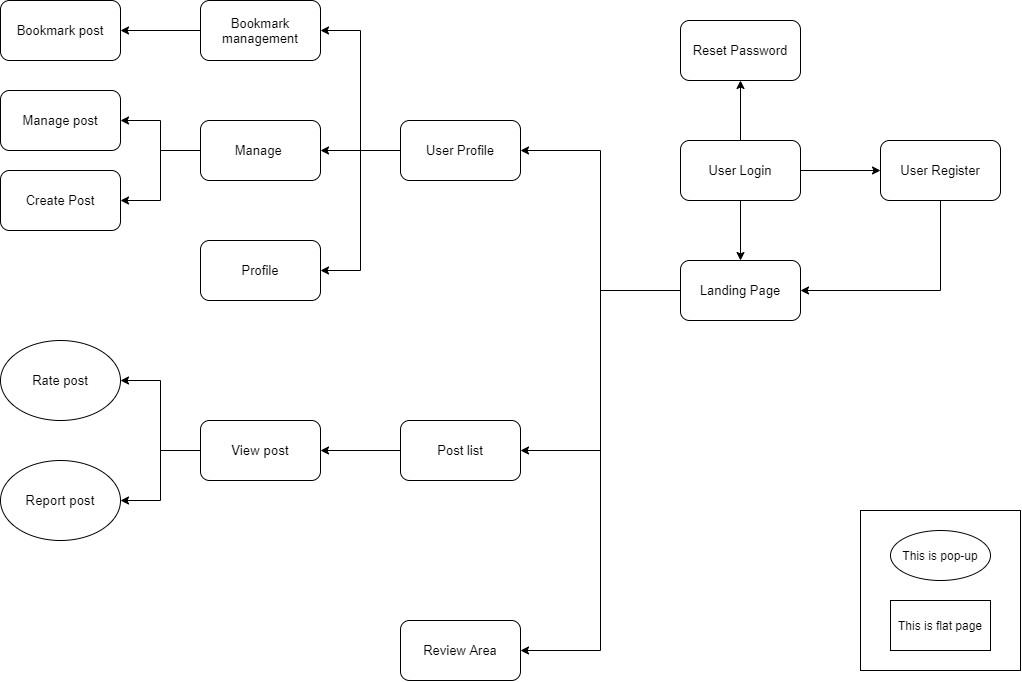
## 1. Introduction

Thuedi is a software system that aims to connect the landlord and potential tenant throughout Hanoi. The Landlord would post the house/flat/land that they want to rent out, and potential renters would look through the list of posts to filter out and possibly find the perfect place for them. The system is expected to be completed in 4 iterations that lasts for a total of 8 weeks.



## 2. System Functions

#### a. Screen Flow



#### b. Screen Details

| **#** | **Feature** | **Screen** | **Description** |
| --- | --- | --- | --- |
| 1 | Authentication | User Login | Login page which allows the user to authenticate into the system using a password and email combination. There will also be a link for registration and password reset. |
| 2 | Authentication | User Register | User registration page for new users. There will be fields detailing basic user properties that they will need to fill out. A separate page can be accessed if the user wants to register as a landlord. |
| 3 | Authentication | Reset Password | Password reset page for if the user forgets their password. The user will enter their email and if the email is valid, they will receive a mail containing a link to reset their password. |
| 4 | Landing Page | Landing Page | There will be a landing page serving as the homepage of the website. The page will contain basic information about the website and the project, as well as links to the other pages. |
| 5 | Post detail | View post | Users will be able to click on a post and be redirected to the post detail page. The post detail page will contain detailed information about the property that the landlord filled in from the post create form. |
| 6 | Post detail | Comment on post | Users will be able to comment on posts so that they can share with other users their experience. |
| 7 | Post detail | Report post | In case there's an inappropriate post, the user will be able to report it to an admin via a report button at the bottom of the post. |
| 8 | Post management | Create Post | If the user is a landlord, they will be able to create posts about their land-for-rent. The create post form will contain detailed information about their land for example: address, number of rooms, number, pictures of the house/flat, other details, etc... |
| 9 | Post management | Update Post | In case there was a mistake in the creation process or if something occurred, the landlord will be able to edit information on their posts. |
| 10 | Post management | Delete Post | The landlord will be able to delete their posts after a accepting a pop-up confirmation. |
| 11 | Posts | Post list | Users will be able to request for a list of posts sorted by relevance. The posts will be displayed as thumbnails with a shortened description. |
| 12 | Review Area | Review Area | There will be a page that contains a list of districts in Hanoi with a thumbnail being a picture of that district. When clicking on the picture, the user will be redirected to a page containing details and a review of that district. |
| 13 | User profile | User profile | Users will be able to manage their profile and update them. |
| 14 | User profile | Bookmark post | The user will be able to add a post to their bookmark list for later viewing. |
| 15 | User profile | Bookmark management | The user's bookmarked posts will be displayed as a paginated list on a separate web page. There will also be buttons for the user to remove their bookmarked items from the list. |

#### c. User Authorization

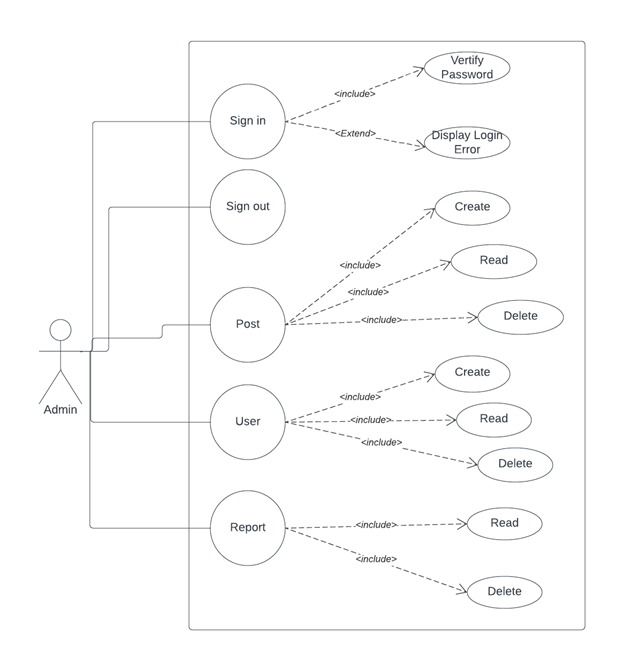
| **Screen** | **Guest** | **User** | **House owner** | **Admin** |
| --- | --- | --- | --- | --- |
| User Login | X | X | X | X |
| User Register | X |  |  | X |
| Reset Password | X | X | X | X |
| Landing Page | X | X | X |  |
| View post | X | X | X |  |
| Comment on post |  | X | X |  |
| Report post |  | X | X |  |
| Create Post |  |  | X |  |
| Update Post |  |  | X |  |
| Delete Post |  |  | X |  |
| Post list | X | X | X |  |
| Review Area | X | X | X |  |
| User profile |  | X | X |  |
| Bookmark post |  | X | X |  |
| Bookmark management |  | X | X |  |
| Admin dashboard |  |  |  | X |
| Admin management report |  |  |  | X |
| Manage account |  |  |  | X |
| Manage post |  |  |  | X |

#### d. Non-Screen Functions

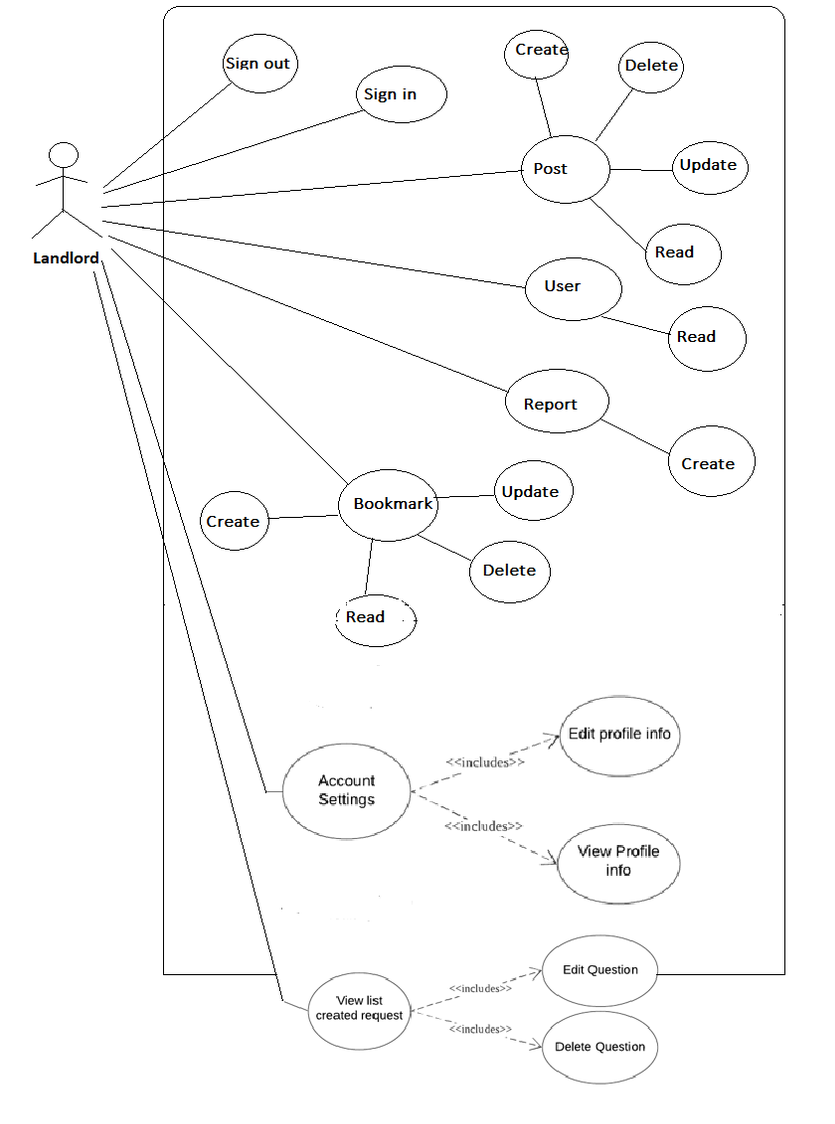
| **#** | **Feature** | **System Function** | **Description** |
| --- | --- | --- | --- |
| 1 | Post detail | View covid status near address | In post details and after filtering by address, there will be information about covid status near the address of the post/filter. |
| 2 | Post detail | View services near address | In post details, there will be a section that displays a map of the area near the post's address. The user will then be able to look around and see what kind of service is near that address. |
| 3 | Post management | Pay to create post | The user's bookmarked posts will be displayed as a paginated list on a separate web page. There will also be buttons for the user to remove their bookmarked items from the list. |
| 4 | Posts | Filter post | Users will be able to filter their post list by property type, general address, price, land area, number of rooms, house direction |
| 5 | Posts | Post list pagination | When the post list is displayed, it will be paginated into 10 items per page. |
| 6 | Posts | Post list breadcrumb | When the post list or post detail is displayed, there will be a breadcrumb section that lists the subsection the current page belongs to |

## 3. **User case**

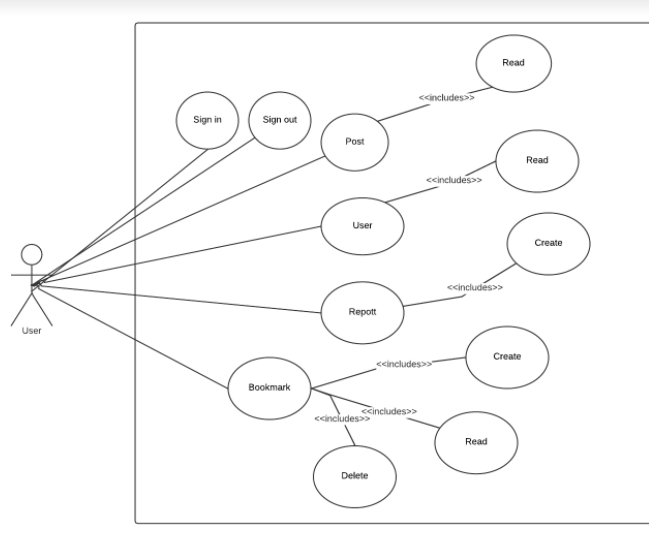
### 3.1. Admin



### 3.2. Landlord



### 3.3. User

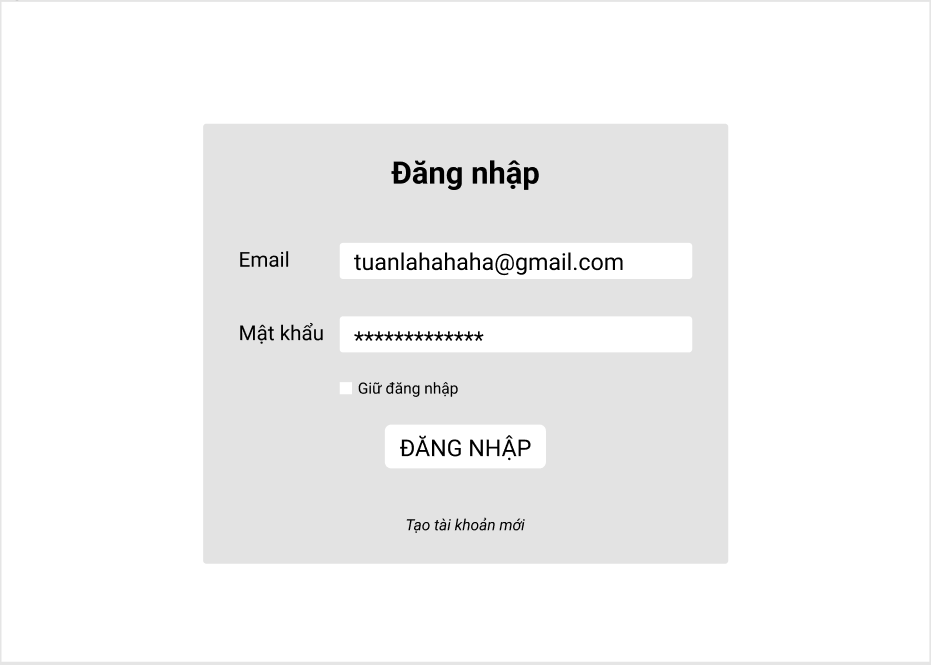


# II. Functional Requirements

## 1. Authentication

### User login

* **Function trigger**: User enters the login screen.
* **Function description**: Login page which allows the user to authenticate into the system using a password and email combination.
* **Function detail**:
* **Screen layout**:



### User registration

* **Function trigger:** User enters the registration page.
* **Function description:** User registration page for new users. There will be fields detailing basic user properties that they will need to fill out. A separate page can be accessed if the user wants to register as a landlord.
* **Function detail:**
* **Screen layout:**

****

### Reset password

* **Function trigger:** User clicks on the forgot password button
* **Function description:** Password reset page for if the user forgets their password. The user will enter their email and if the email is valid, they will receive a mail containing a link to reset their password.
* **Function detail:** An email will be sent to the user’s email address, containing a one-time-use password reset link unique to the user. The user will then enter their new password on a form.

## 2. Landing page

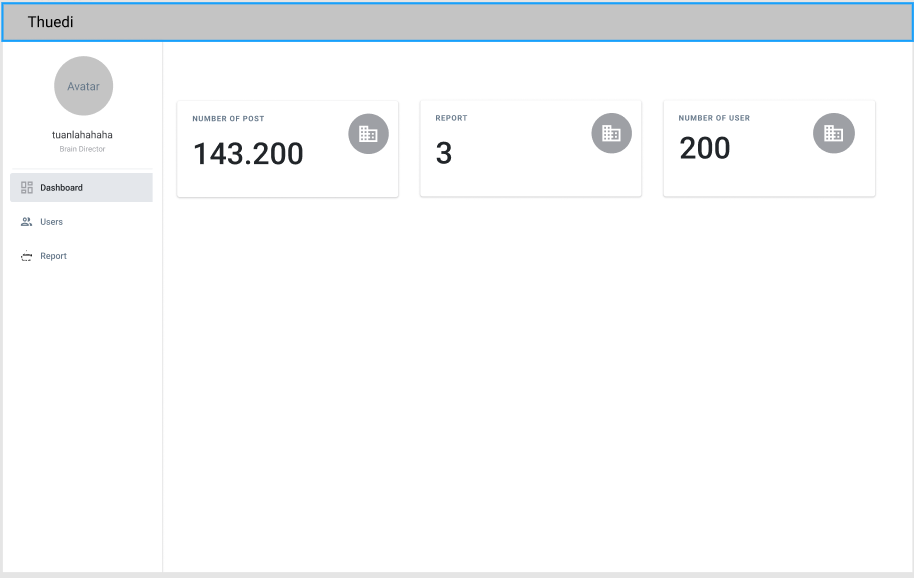
### Landing page

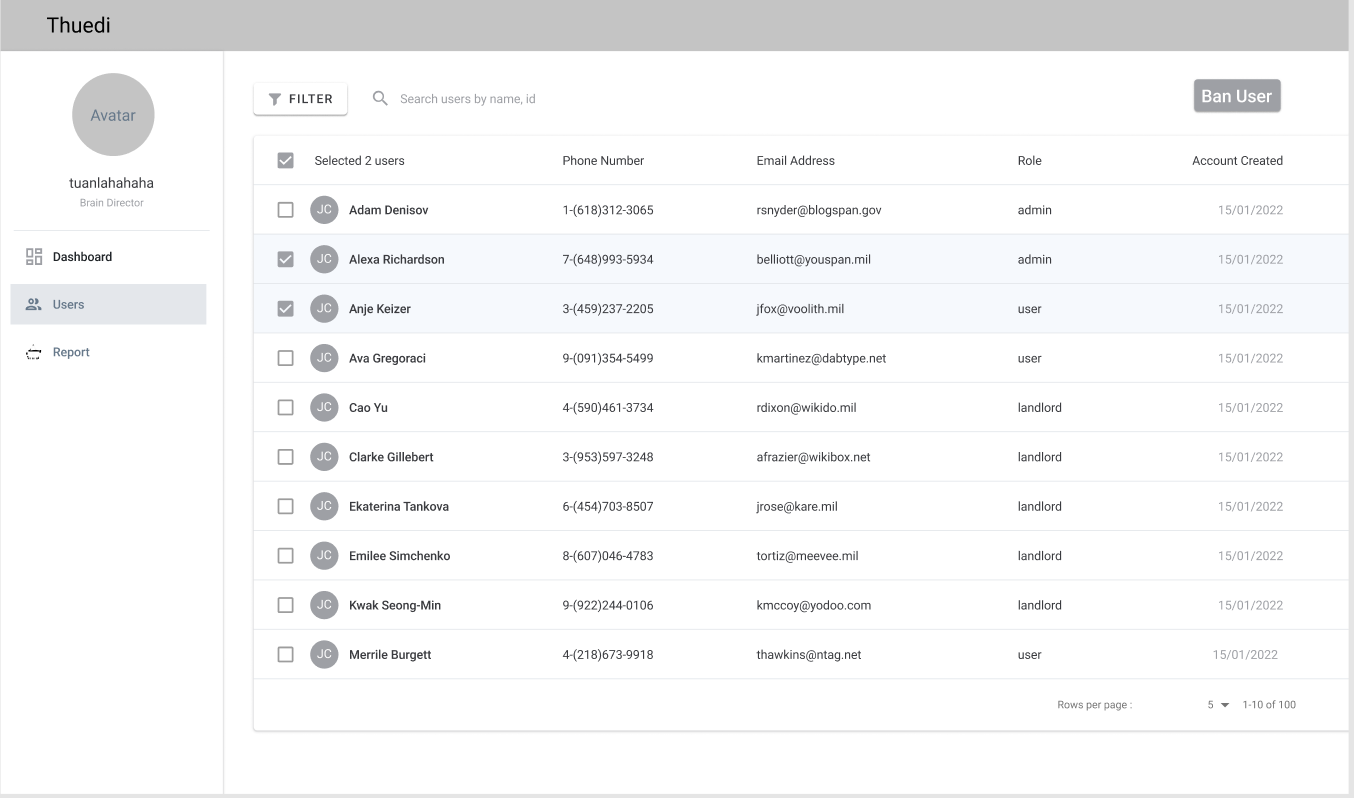
* **Function trigger:** User enters the website through the index page.
* **Function description:** There will be a landing page serving as the homepage of the website. The page will contain basic information about the website and the project, as well as links to the other pages.
* **Function detail:** The page will display some basic information about the website and project like: What the website is about; Who are the founders; What can the user do on the website; FAQ; etc… The page will also contain promoted posts and other information.

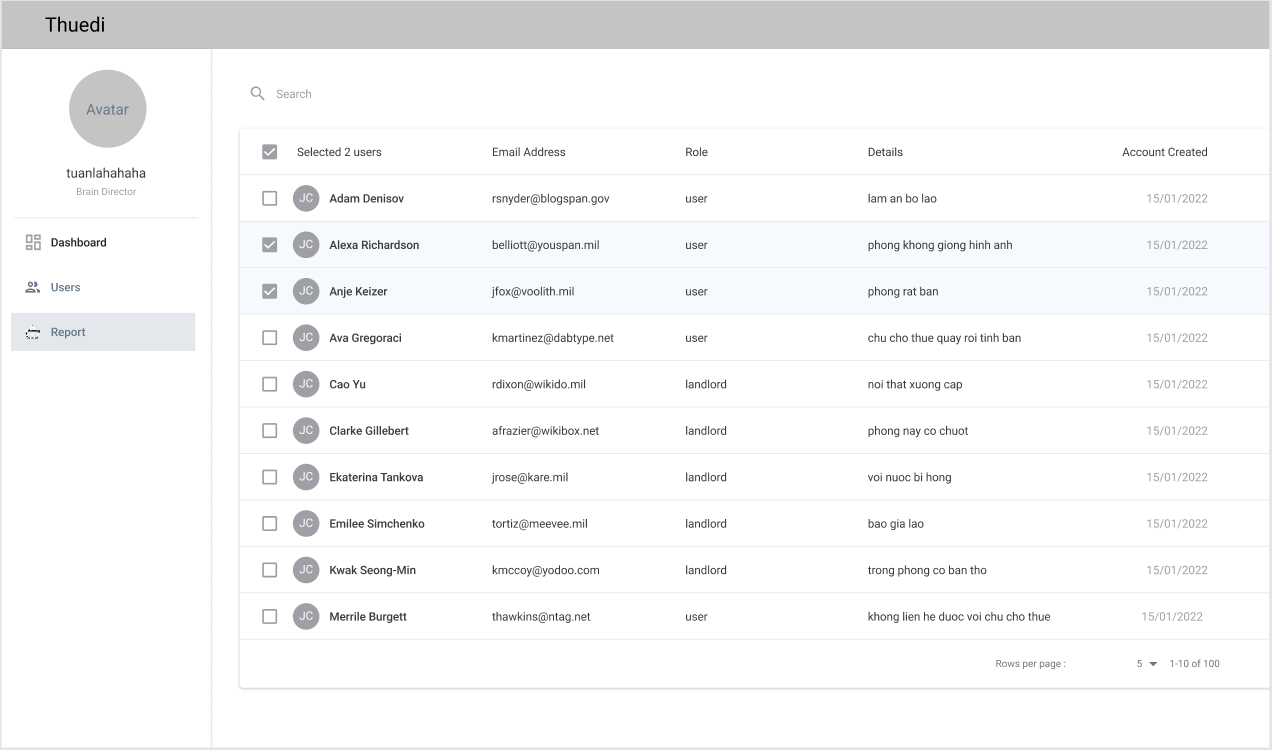
## 3. Management

### Admin page

* **Function trigger**: A user with the admin role enters the admin page.
* **Function description**: The admin page will be used to manage everything on the website. It will contain information about every post/user, along with functionalities to manage them. Reports from the user will also appear here.
* **Function detail**: Users, posts, reports will be queried from the database and displayed here. The admin can then do CRUD functions on them.
* **Screen layout:**

****

****

****

## 4. Post detail

### View Post

* **Function trigger**: User enters post detail.
* **Function description**: Users will be able to click on a post and be redirected to the post detail page. The post detail page will contain detailed information about the property that the landlord filled in from the post create form.
* **Function detail**: Information about the post will be queried from multiple tables on the database and put together in the post detail page.
* **Screen layout:**

****

### View covid status near address

* **Function trigge**r: User enters post detail page.
* **Function description**: In post details and after filtering by address, there will be information about covid status near the address of the post/filter.
* **Function detail**: An external API will be called with the post’s address as the payload. The API will then return the covid information about the district the address belongs to.

### Comment on post

* **Function trigger**: Authenticated users post a comment from the comment form.
* **Function description**: Users will be able to comment on posts so that they can share with other users their experience.
* **Function detail**: When the user submits their comment, it will be inserted into the database and displayed to other users once they refresh their page.

### View services near address

* **Function trigger**: User enters post detail page.
* **Function description**: In post details, there will be a section that displays a map of the area near the post's address. The user will then be able to look around and see what kind of service is near that address.
* **Function detail**: An external API will be called with the post’s address as the payload. The API will then render a map with the address at the center and services/interest points around it.

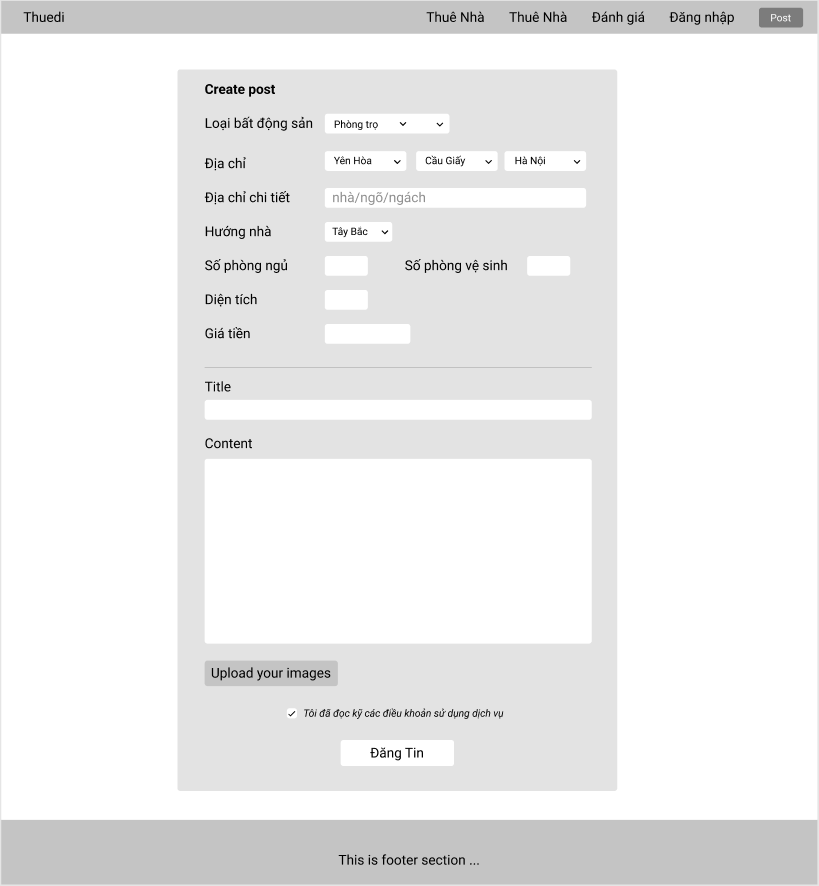
### Report post

* **Function trigger**: User enter icon to report post
* **Function description**: In case there's an inappropriate post, the user will be able to report it to an admin via a report button at the bottom of the post.
* **Function detail**: Once the report button has been clicked, the post will be flagged and displayed on the admin page. The admin will then review the report and determine what will happen to the post.

## 5. Post management

### Create post

* **Function trigger:** in navigation bar.
* **Function description:** If the user is a landlord, they will be able to create posts about their land-for-rent. The create post form will contain detailed information about their land for example: address, number of rooms, number, pictures of the house/flat, other details, etc...
* **Function detail:** After the landlord fills in the create post form and clicks submit, the payment screen will appear. After the payment is successfully processed, the post will appear in the post list.
* **Screen layout:**

****

### Pay to create post

* **Function trigger:** The create post form is submitted.
* **Function description:** The user's bookmarked posts will be displayed as a paginated list on a separate web page. There will also be buttons for the user to remove their bookmarked items from the list.
* **Function detail:** The payment page will display payment information for the landlord. Once the payment is completed, the create post process is over.

### Update post

* **Function trigger:** User clicks on the update post button.
* **Function description:** In case there was a mistake in the creation process or if something occurred, the landlord will be able to edit information on their posts.
* **Function detail:** A form similar to the create post form will appear with pre-filled values from the post that is being edited. The landlord can then change each values then submit the form.

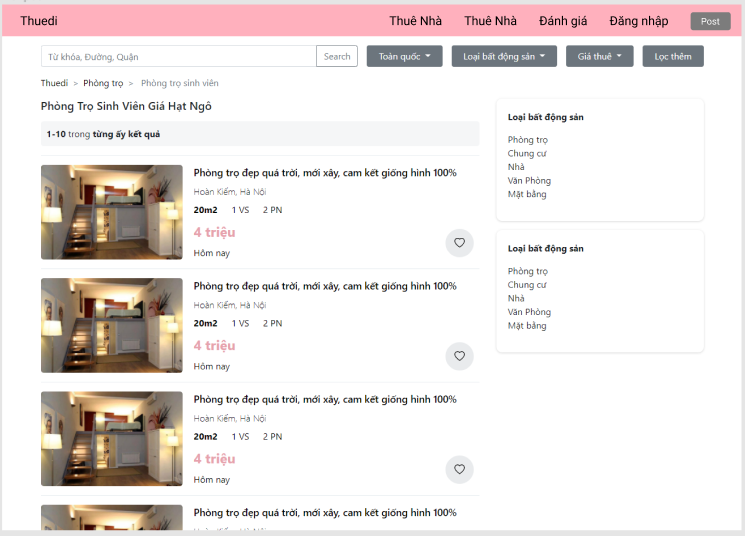
### Delete post

* **Function trigger:** User clicks on the delete post button.
* **Function description:** The landlord will be able to delete their posts after accepting a pop-up confirmation.
* **Function detail:** When the user wants to delete their post, click on the delete button and a dialog will appear asking if they are sure they want to delete it, if they press OK, the post will be deleted, if not, go back to the post management.

## 6. Posts

### Post list

* **Function trigger:** go to post list page on navigation bar or landing page.
* **Function description:** Users will be able to request for a list of posts sorted by relevance. The posts will be displayed as thumbnails with a shortened description.
* **Function detail:** The server will return posts based on the current page number. Those posts will then be displayed with a thumbnail and shortened description format.
* **Screen layout:**

****

### Filter post

* **Function trigger:** Click on a filter dropdown list.
* **Function description:** Users will be able to filter their post list by property type, general address, price, land area, number of rooms, house direction
* **Function detail:** When a user clicks on a filter dropdown list, the posts page will display the list of room types that the user wants.

### Post list pagination

* **Function trigger:** at the bottom of the post details page, click on pagination.
* **Function description:** When the post list is displayed, it will be paginated into 10 items per page.
* **Function detail:** helps the user to know his current location on the website, and when clicking on the location, the user can return to that location.

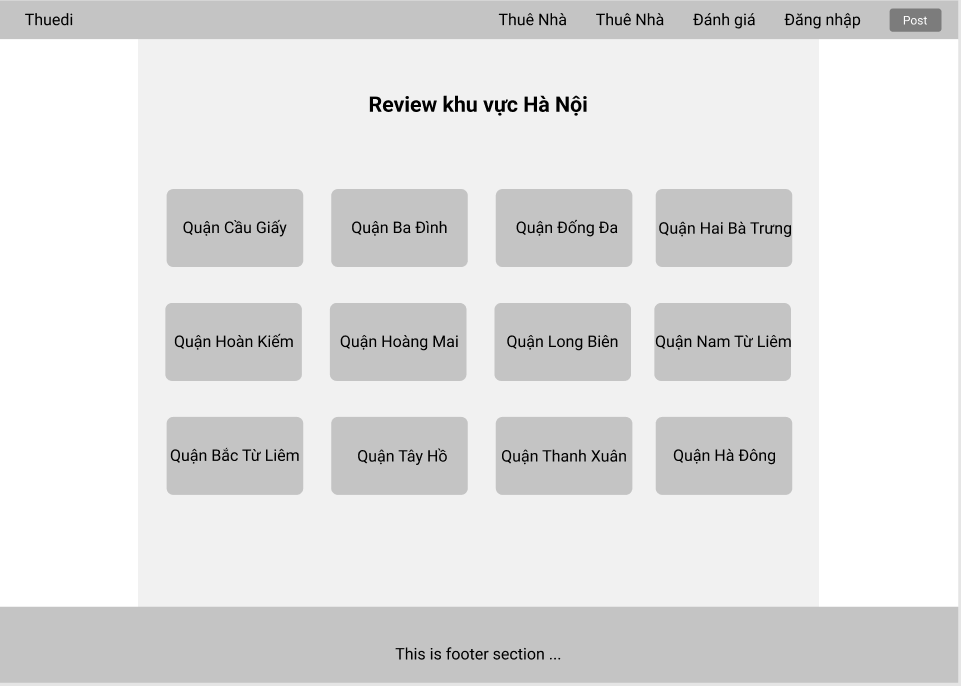
### Post list breadcrumb

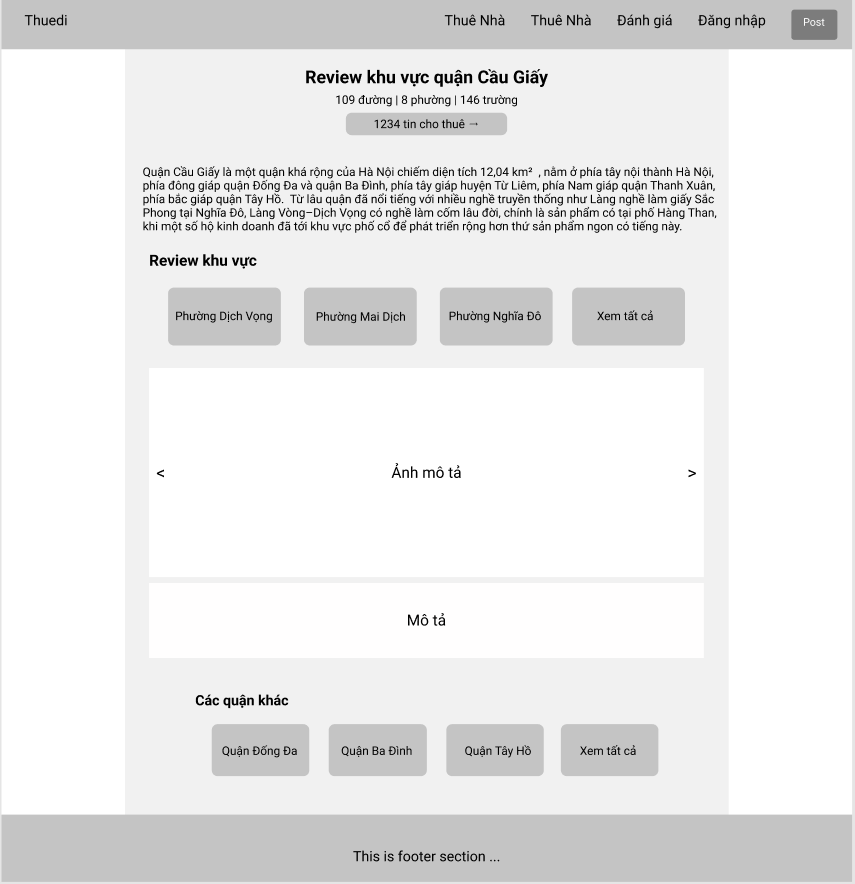
* **Function trigger:** at the top of the post list page, click on breadcrumb.
* **Function description:** When the post list or post detail is displayed, there will be a breadcrumb section that lists the subsection the current page belongs to.
* **Function detail:** a breadcrumb or breadcrumb trail is a graphical control element used as a navigational aid in user interfaces and on web pages. It allows users to keep track and maintain awareness of their locations within programs, documents, or websites**.**

**7. Review area**

### Show a review area

* **Function trigger:** In post detail or navigation bar.
* **Function description:** Review area page helps users know the information of that area.
* **Function detail:** There will be a page that contains a list of districts in Hanoi with a thumbnail being a picture of that district. When clicking on the picture, the user will be redirected to a page containing details and a review of that district.
* **Screen layout:**

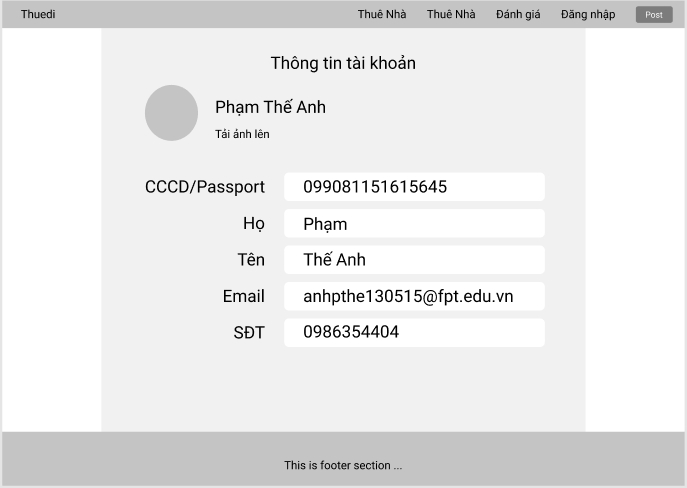
****

****

## 8. User profile

### User Profile

* **Function trigger:** User enters the user profile screen.
* **Function description:** Users will be able to manage their profile and update them.
* **Function detail:** The page displays the user's personal information such as avatar, name, phone number, email,... and allows the user to edit that information.
* **Screen layout:**

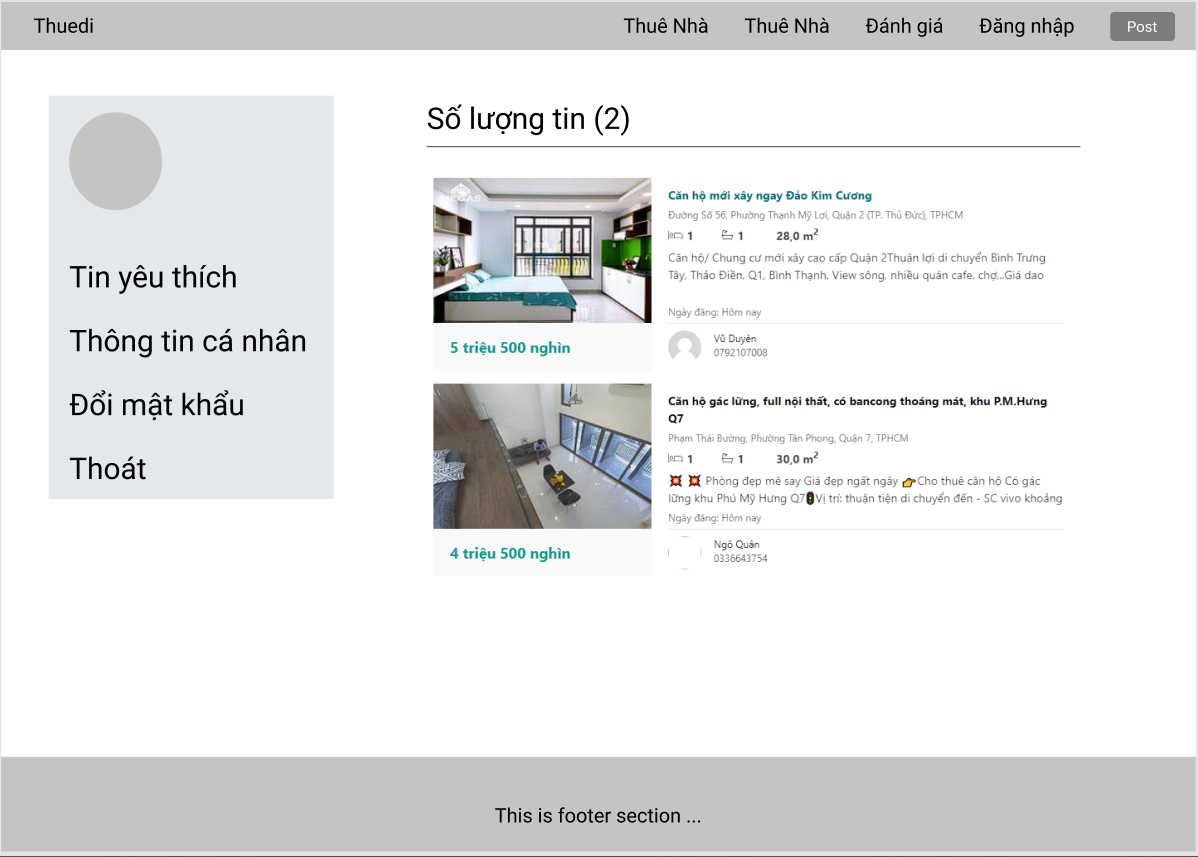
****

### Bookmark post

* **Function trigger:** User clicks on bookmark button in post detail.
* **Function description:** The user will be able to add a post to their bookmark list for later viewing.
* **Function detail:** When viewing a post that the user wants to save, the user can click on the bookmark icon to save it to their bookmarks list.

### Bookmark management

* **Function trigger:** User enters the bookmark management screen.
* **Function description:** The user's bookmarked posts will be displayed as a paginated list on a separate web page. There will also be buttons for the user to remove their bookmarked items from the list.
* **Function detail:** The page displays the user bookmark list, the user can view and remove their bookmarked items from the list.
* **Screen layout:**

****

# III. Non-function requirement

## 1 External Interfaces

### 1.1.User interfaces

UI-1: The web application display shall conform to the user and provide a friendly interface.

UI-2: The font size is easy to distinguish where is the title, where is the content.

UI-3: Consistent interface: font size, font, color, the logo must be in sync.

### 1.2.Software interfaces

N/A

## 1.3.Hardware interfaces

No hardware interfaces have been identified. Our system is aimed at building software systems, so the system has no hardware.

## 2 Quality Attributes

### 2.1 Usability

USA-1: The steps and flows are short and easy to understand so that users can interact with the application without training.

USA-2: The user interface is clean, friendly, and consistent from the layout, component to the button, text field.

USA-3: The input action is flexible with both keyboard and mouse.

USA-4: Error and warning messages are simple.

### 2.2 Reliability

N/A

### 2.3 Performance

● Reduce as less as possible query to database in loop

● Resource utilization: use very few computer resources such as ram, memory, CPU.

### 2.4 Dependence

#### 2.4.1. Security

- Only the Admin account has access to system management.

- The system allow users to register for an account on the system with other roles as user and landlord.

#### 2.4.2 Safety

- The password of the account is encrypted since creating a moment.

- The password should be hidden in the client.

#### 2.5 Supportability

- The account's password must be encrypted

- Named according to the Java Convention

- Project structure according to MVC model